Applications of Fuzzy Logic to Graph Theory

M. Blue,[*] B. Bush,[†] and J. Puckett[‡]
Energy and Environmental Analysis Group
Technology and Safety Assessment Division
Los Alamos National Laboratory
15 August 1997

Abstract

Graph theory has numerous applications to problems in systems analysis, operations research, transportation, and economics. In many cases, however, some aspects of the graph-theoretic problem are uncertain. In these cases, it can be useful to deal with this uncertainty using the methods of fuzzy logic. This paper discusses the taxonomy of fuzzy graphs, formulates some standard graph-theoretic problems (shortest paths, maximum flow, minimum cut, and articulation points) in terms of fuzzy graphs, and provides algorithmic solutions to these problems, with examples.

Keywords: graph theory, shortest paths, maximum flow, minimum cut, articulation points, fuzzy sets, fuzzy logic, fuzzy graph.

[*] *Mailing address:* Mathematics Department, University of Texas, Austin, TX 78712 USA (mblue@math.utexas.edu)

[†] *Mailing address:* Energy and Environmental Analysis Group, Technology and Safety Assessment Division, Los Alamos National Laboratory, Los Alamos, NM 87545 USA (bwb@lanl.gov)

[‡] *Mailing address:* Science Applications International Corporation, 4301 N. Fairfax Drive, Arlington, VA 22203 USA (jpucket@std.saic.com)

I. Introduction

Graph theory has numerous applications to problems in systems analysis, operations research, transportation, and economics. In many cases, however, some aspects of a graph-theoretic problem may be uncertain. For example, the vehicle travel time or vehicle capacity on a road network may not be known exactly. In such cases, it is natural to deal with the uncertainty using the methods of fuzzy logic. This paper presents a taxonomy of fuzzy graphs, providing a catalog of the various types of "fuzziness" possible in graphs. We also formulate some standard graph-theoretic problems (shortest paths, maximum flow, minimum cut, and articulation points) in terms of fuzzy graphs, and provide algorithmic solutions to these problems, with examples.

Several other formulations of fuzzy graph problems have appeared in the literature. Klein [Kl 91] discusses a number of alternative methods for assigning membership grades to paths in a graph. Lin and Chern [LC 93] treat the shortest path problem in terms of a fuzzy linear program. Specialized applications to PERT/CPM and decision trees are provided by Chanas and Kamburowski [CK 81], Itakura and Nishikawa [IN 84], and Adamo [Ad 80]. Chanas and Kolodziejczyk [CK 82; CK 84; CK 86] consider crisp flows as solutions to the maximum flow on a graph with fuzzy edge capacity constraints. Peng and Juang [PJ 93] construct flow membership grades for maximum flows. Finally, Kim and Roush [KR 82] examine the problem of Boolean flows on a fuzzy network. The approach presented in this paper is distinguished by its uniform application of several key guiding principles—the construction of fuzzy graph membership grades via the ranking of fuzzy numbers, the preservation of membership grade normalization, and the "collapsing" of fuzzy sets of graphs into fuzzy graphs—to the classic shortest path, maximum flow, minimum cut, and articulation point problems.

In the rest of this section we introduce the notation for fuzzy sets used in this paper. References [Ka 86; KF 88; KG 85] provide additional background.

A. Fuzzy Sets

A *fuzzy set* is a set where there is some measure of uncertainty of membership in the set. For a fuzzy set S, each element of a referential set Ω must be assigned a membership in S:

(1) $$\mu_S : \Omega \to M,$$

where μ_S is the *membership function* for the set and M is the set of allowed measurements. Typically M is chosen to be the unit interval, $[0,1]$, so that

(2) $$\mu_S : \Omega \to [0,1].$$

One also usually requires that the measure μ_S be *normalized*: namely,

(3) $$\exists x \in \Omega \text{ such that } \mu_S(x) = 1.$$

Other choices for M and the normalization condition are possible. In this paper we consider measures satisfying Equations (2) and (3).

A *crisp*, or non-fuzzy, set can be treated as a fuzzy set whose measure attains only the values unity and zero:

(4) $$\mu_S(x) \in \{0,1\} \text{ for } \forall x \in \Omega .$$

Fuzzy numbers are fuzzy sets where the referential set Ω is the set of real numbers, integers, etc. The interpretation here is that the belief in a fuzzy number A being any particular number x is given by its measure at the number, $\mu_A(x)$. The appendix of this paper discusses fuzzy numbers in more detail.

B. Supports and Level Sets

The *support* of a fuzzy set S, written as $\mathrm{supp}(S)$, is the crisp subset of the referential set Ω defined by

(5) $$\mathrm{supp}(S) = \left\{ x \in \Omega \middle| \mu_S(x) > 0 \right\}.$$

The interpretation of this is that the support of a fuzzy set is the set of objects that are possibly in the set.

The α *-cut* of a fuzzy set S, denoted by S_α, is the crisp subset of Ω that contains all of the elements of S with at least the given degree of membership α:

(6) $$S_\alpha = \left\{ x \in \Omega \middle| \mu_S(x) \geq \alpha \right\}.$$

Similarly, the α *-level cut* of a fuzzy set S, denoted by S^α, is the crisp subset of Ω that contains all of the elements of S with exactly the given degree of membership α:

(7) $$S^\alpha = \left\{ x \in \Omega \middle| \mu_S(x) = \alpha \right\}.$$

Hence, $S^\alpha \subseteq S_\alpha$. The support can be rewritten in terms of cuts as

(8) $$\mathrm{supp}(S) = S_{0+} = \bigcup_{\alpha \in (0,1]} S^\alpha .$$

Note that $S^\alpha = S_\alpha$ if and only if $\alpha = 1$. The α -cuts and α -level cuts provide a means of treating the elements of the set at specified levels of belief.

The *level set* of S, denoted by Λ_S, is a subset of [0,1] containing the values α that determine distinct α -cuts: explicitly,

(9) $$\Lambda_S = \left\{ \alpha \in [0,1] \middle| \mu_S(x) = \alpha \text{ for some } x \in \Omega \right\}.$$

C. *Fuzzy Functions*

It is useful to define functions on fuzzy sets. Any unary function or operation

(10) $$f : D \to R$$

can be generalized to apply to fuzzy sets. Consider a fuzzy set A with measure $\mu_A : D \to [0,1]$; we define the measure for $f(A)$ as

(11) $$\mu_{f(A)}(y) = \sup_{y=f(x)} \{\mu_A(x)\},$$

where $\mu_{f(A)} : R \to [0,1]$. The supremum function in Equation (11) guarantees the preservation of the normalization condition. Likewise, one can generalize any binary function or operation as

(12) $$\mu_{A \otimes B}(z) = \sup_{x \otimes y = z} \{\min\{\mu_A(x), \mu_B(y)\}\}.$$

Here the function $\otimes$ could be, for example, an operation from set theory (e.g., $\otimes \in \{\cup, \cap, -\}$) or—in the case of fuzzy numbers—an arithmetical operation (e.g., $\otimes \in \{+, -, \times, \div, \min, \max\}$), etc. Once again, the supremum function in Equation (12) maintains the normalization condition: there is always at least one combination of x and y which have unit measure, so the function's measure will be unity for at least one z.

We can also generalize the comparison operators in the same way as other arithmetic operations if we interpret them as Boolean-valued functions:

(13) $$\otimes : \Re \times \Re \to \{\text{true}, \text{false}\},$$

where $\otimes \in \{=, <, >, \neq, \geq, \leq\}$. If we use the convenient shorthand notation,

(14a) $$\hat{\mu}_{A \otimes B} \equiv \mu_{A \otimes B}(\text{true}),$$

(14b) $$\overline{\mu}_{A \otimes B} \equiv \mu_{A \otimes B}(\text{false}),$$

we can express the complementary relationship between =, <, and > versus ≠, ≥, and ≤, respectively, as

(15) $$\hat{\mu}_{A \otimes B} = \overline{\mu}_{A \overline{\otimes} B};$$

for example,

$$\hat{\mu}_{A > B} = \overline{\mu}_{A \leq B} \quad \text{and} \quad \hat{\mu}_{A \leq B} = \overline{\mu}_{A > B}.$$

It can be shown that the normalization condition translates to

(16) $$\hat{\mu}_{A \otimes B} = 1 \text{ or } \hat{\mu}_{A \overline{\otimes} B} = \overline{\mu}_{A \otimes B} = 1;$$

for example,

$$\hat{\mu}_{A = B} = 1 \text{ or } \hat{\mu}_{A \neq B} = 1.$$

Note that there is no other general relationship between $\hat{\mu}_{A\otimes B}$ and $\overline{\mu}_{A\otimes B}$. Other approaches to the problem of ranking fuzzy numbers have appeared in References [BD 85; Ch 85; DP 83; DVV 88; OM 87; Ov 89]. We adopt the one in Equation (14) because it provides memberships for ranking based on the fundamental definition given by Equation (12). All ranking procedures suffer from a certain awkwardness in interpretation and unsatisfactory application to specific cases. In the case of Equation (14), this arises when $\mu(x)$ is continuous around a point $x*$ for which $\mu_A(x*) = \mu_B(x*) = 1$, so that $\hat{\mu}_{A=B} = \hat{\mu}_{A\neq B} = \hat{\mu}_{A<B} = \hat{\mu}_{A\leq B} = 1$.

II. Fuzzy Graphs

In this section we outline the notation we use for fuzzy graphs and provide a classification of different types of graph fuzziness.

A. Notation

We use the following notation to describe graphs [Gi 85]. Only directed graphs are treated here—undirected graphs are handled as a special case of digraphs. At this point, we do not make a distinction between *crisp graphs* and *fuzzy graphs*.

A *graph* G consists of a set of *vertices* V and a set of *edges* E:

(17) $$G = (V, E).$$

We label the vertices and edges with indices:

(18a) $$V = \{v_1, v_2, \ldots, v_{n_V}\},$$

(18b) $$E = \{e_1, e_2, \ldots, e_{n_E}\},$$

where n_V is the number of vertices and n_E is the number of edges. Each edge has a *head* and a *tail*:

(19a) $$h_i = \mathrm{head}(e_i),$$

(19b) $$t_i = \mathrm{tail}(e_i).$$

In a *weighted graph*, each edge also has a *weight* (sometimes called its *length* or *capacity*),

(20) $$w_i = W(e_i),$$

specified by a weight function W that maps edges to numbers (which may be crisp or fuzzy).

A *path* P is a sequences of edges

(21) $$P = (e_{i_1}, e_{i_2}, \ldots, e_{i_n}),$$

where the head of one edge is the same as the tail of the following edge,

(22) $$h_{i_k} = t_{i_{k+1}} \text{ for } k = 1,\ldots,(n-1)\,.$$

The *head* of the path is h_{i_n} and the *tail* of the path is t_{i_1}:

(23a) $$h_{i_n} = \mathrm{head}(P)\,,$$

(23b) $$t_{i_1} = \mathrm{tail}(P)\,.$$

If the graph is weighted, the path has a *length* given by the sum of the weights for the edges in the path,

(24) $$\ell_P = \mathrm{length}(P) = \sum_{e_k \in P} w_k\ .$$

A flow F for a graph assigns a number to each edge of that graph,

(25) $$f_i = F(e_i)\,,$$

subject to the condition that the flow on each edge is non-negative,

(26a) $$0 \le f_i \text{ for } i = 1,\ldots,n_E\,,$$

not more than the capacity of the edge,

(26b) $$f_i \le w_i \text{ for } i = 1,\ldots,n_E\,,$$

and conserved at the vertices,

(26c) $$\sum_{\substack{j=1,\ldots,n_E \\ h_j = v_i}} f_j = \sum_{\substack{j=1,\ldots,n_E \\ t_j = v_i}} f_j \text{ for } i = 1,\ldots\hat{a},\ldots,\hat{b},\ldots,n_V\,,$$

except at the *source* and *sink* vertices, v_a and v_b, respectively. Note that special interpretation will be required for Equation (26) when the weights w_i or flows f_i are fuzzy numbers. The *value* of a flow can be measured at the source or at the sink:

(27) $$\mathrm{val}(F) = \sum_{\substack{j=1,\ldots,n_E \\ t_j = v_a}} f_j = \sum_{\substack{j=1,\ldots,n_E \\ h_j = v_b}} f_j\ .$$

A *cut* K is a set of edges that disconnects the sink vertex v_b from the source vertex v_a — i.e., there is no path in the graph from v_a to v_b that does not include an edge in K. For a weighted graph, the *value* of the cut is the sum of the weights in the cut:

(28) $$\mathrm{val}(K) = \sum_{e_k \in \mathrm{forward}(K)} w_k\ ,$$

where $\mathrm{forward}(K)$ is the set of *forward* edges in the cut (i.e., edges from the source's partition to the sink's partition). A *proper cut* is a cut with no proper subsets that are also cuts, and an *improper cut* is a cut with a proper subset that is also a cut.

At this point, we would also like to introduce a shorthand notation [KF 88]:

(29) $$A = A_1 \setminus \mu_1 + A_2 \setminus \mu_2 + A_3 \setminus \mu_3 + \cdots,$$

which means

(30) $$\mu_A(x) = \begin{cases} \max\limits_{A_i = x}\{\mu_i\} & \text{when } x = A_i \text{ for some } i \\ 0 & \text{otherwise} \end{cases}.$$

This makes it convenient to express the types of graph fuzziness discussed in the next section.

B. Taxonomy of Graph Fuzziness

There are several ways in which a graph can be fuzzy. Below we classify the primary types of fuzziness possible in graphs.

1. Type I: Fuzzy Set of Crisp Graphs

A trivial type of graph fuzziness arises from considering a fuzzy set G of crisp graphs G_i:

(31) $$G = G_1 \setminus \mu_1 + G_2 \setminus \mu_2 + \cdots + G_{n_G} \setminus \mu_{n_G}.$$

From the point of view of analysis, this type of fuzziness is really not very interesting unless the graphs G_i have some vertices or edges in common. Even when the basic crisp graphs have vertices or edges in common, analysis is difficult unless the commonality has a regular structure. The case of most interest, which we will call *Type I'*, occurs when each of the crisp graphs G_i has the same set of vertices:

(32) $$V = V_1 = V_2 = \cdots = V_{n_G}.$$

Thus it is the presence and configuration of the edges that is fuzzy for these graphs. Several other variations of Type I graph fuzziness exist, but we will not catalog them all here.

The question of interpretation of fuzziness inevitably arises. Here are two possible scenarios involving this type of fuzziness:

> *Type I:* You would like to make some changes to your house's electrical system. However, the builder has mixed up the records for all the houses on the block. Thus, you can get a copy of the electrical systems for all the houses on your block, but there is no way to distinguish which electrical plan corresponds to which house.
>
> *Type I':* You have been given two maps with which to plan the shortest automobile route from one city to another. The two maps are of different dates and thus have different road networks. Unfortunately, there is no indication which map is more recent.

2. Type II: Crisp Vertex Set and Fuzzy Edge Set

It may happen that a graph has known vertices, but unknown edges. In this case the vertex set is crisp and the edge set is fuzzy:

(33a) $$V = \{v_1, v_2, \ldots, v_{n_V}\} ,$$

(33b) $$E = e_1 \setminus \mu_1 + e_2 \setminus \mu_2 + \cdots + e_{n_E} \setminus \mu_{n_E} ,$$

where each edge e_i is crisp (i.e., it has fixed head, tail, and weight). Here is a possible scenario involving this type of graph fuzziness:

> *Type II:* You have to plan the shortest automobile route from one city to another. Unfortunately, there is a lot of road construction taking place, so some roads may be closed, but it is not known with certainty which roads are affected.

3. Type III: Crisp Vertices and Edges with Fuzzy Connectivity

In contrast with Type II graph fuzziness, it may occur that the graph has known vertices and edges, but unknown edge connectivity. Here both the vertex and edge sets are crisp, but the edges themselves have fuzzy heads and tails:

(34a) $$V = \{v_1, v_2, \ldots, v_{n_V}\} ,$$

(34b) $$E = \{e_1, e_2, \ldots, e_{n_E}\} ,$$

(34c) $$h_i = h_{i,1} \setminus \sigma_{i,1} + h_{i,2} \setminus \sigma_{i,2} + \cdots + h_{i,n_V} \setminus \sigma_{i,n_V} \text{ for } i = 1, \ldots, n_E ,$$

(34d) $$t_i = t_{i,1} \setminus \tau_{i,1} + t_{i,2} \setminus \tau_{i,2} + \cdots + t_{i,n_V} \setminus \tau_{i,n_V} \text{ for } i = 1, \ldots, n_E .$$

This type of graph fuzziness is relevant in the following example:

> *Type III:* You have to plan the shortest automobile route from one city to another. Many of the routes involve ferry crossings over a large body of water. Unfortunately, the ferry schedule is vague as to which drop-off points correspond to which pick-up points.

4. Type IV: Fuzzy Vertex Set and Crisp Edge Set

In an analogy with Type II graph fuzziness, it may happen that a graph has unknown vertices, but known edges. In this case the vertex set is fuzzy and the edge set is crisp:

(35a) $$V = v_1 \setminus \mu_1 + v_2 \setminus \mu_2 + \cdots + v_{n_V} \setminus \mu_{n_V} ,$$

(35b) $$E = \{e_1, e_2, \ldots, e_{n_E}\} .$$

Equation (35) requires careful interpretation because edges cannot exist in a graph if their head and tail vertices do not exist; we call the edge set crisp even though it depends on fuzzy vertices. The following example shows how this type of graph fuzziness might occur:

Type IV: You would like to give presentations at several conferences consecutively and need to determine the most cost-effective travel plan for attending the conferences. The conference committees, however, have not yet revealed the locations of their respective conferences.

5. Type V: Crisp Graph with Fuzzy Weights

A fifth type of graph fuzziness—one of much interest—occurs when the graph has known vertices and edges, but unknown weights (or capacities) on the edges. Thus only the weights are fuzzy:

(36) $$w_i = w_{i,1} \setminus \mu_{i,1} + w_{i,2} \setminus \mu_{i,2} + \cdots \ .$$

Here is a possible scenario involving this type of graph fuzziness:

Type V: You have to plan the quickest automobile route from one city to another. Unfortunately, the map gives distances, not travel times, so you do not know exactly how long it takes to travel any particular road segment.

6. Relationship between Types of Graph Fuzziness

It is clear that a fuzzy graph may have various combinations of fuzziness of types I–V. Also, the five types of fuzziness discussed above are somewhat interrelated in that it is possible in some cases to *collapse* the fuzziness of one type into another type. For example, a graph with Type I' fuzziness can be converted to a graph with Type II fuzziness by identifying the correspondence between edges in the different Type I' graph elements:

(37a) $$E^{\mathrm{II}} = e_1^{\mathrm{II}} \setminus \lambda_1 + e_2^{\mathrm{II}} \setminus \lambda_2 + \cdots + e_{n_E^{\mathrm{II}}}^{\mathrm{II}} \setminus \lambda_{n_E^{\mathrm{II}}} \ ,$$

where

(37b) $$\lambda_i = \max_j \left\{ \mu_j \;\middle|\; G_j \text{ contains an edge from } t_i^{\mathrm{II}} \text{ to } h_i^{\mathrm{II}} \right\}.$$

This procedure always preserves the normalization condition. One can also construct other schemes for collapsing fuzziness; however, not all schemes preserve the normalization condition.

It is also possible to *expand* the fuzziness of a graph. Type II fuzzy graphs can be expanded to Type I' fuzzy graphs by enumerating all of the possible crisp graphs consistent with the fuzzy graph: simply assign a membership

(38) $$\mu_j = \min_{e_i \in E_j} \{ \eta_i \},$$

where η_i is the membership of edge i in the Type II fuzzy graph, for each Type I' edge set E_j in the power set $\wp(\{e_1, e_2, \ldots, e_{n_E}\})$ of possible edge sets. A similar procedure is available for expanding Type III or IV graphs into Type I' graphs.

C. Other Fuzzy Graph-Theoretic Constructs

Our general approach for defining fuzzy graph-theoretic objects such as paths and flows on a fuzzy graph G is to assign a membership to the object based on the minimum memberships of crisp objects over the components of G.

1. Fuzzy Paths

We define a *fuzzy path* with a tail vertex v_a and a head vertex v_b on a graph G to be a Type II fuzzy graph with edge memberships $\mu(i)$ such that

(39) $$\mu(i) > 0 \text{ implies } \exists P \in \Pi_{ab} \text{ such that } \mu(j) \geq \mu(i), \forall e_j \in P$$

and

(40) $$\exists P \in \Pi_{ab} \text{ such that } \mu(j) = 1, \forall e_j \in P ,$$

where Π_{ab} is the set of crisp paths from v_a to v_b. Equations (39) and (40) induce a membership for paths $P \in \Pi_{ab}$ in the fuzzy path:

(41) $$\pi(P) = \min_{e_i \in P}\{\mu(i)\}.$$

The normalization condition ensures that there will always be at least one *most likely path* along which $\mu(i) = 1$. If the graph is weighted, then the *fuzzy length* of the path is defined as the fuzzy number with membership

(42) $$\lambda(x) = \max_{\substack{P\in\Pi_{ab} \\ x=\text{length}(P)}} \left\{ \min_{e_i \in P}\{\mu(i)\} \right\} = \max_{\substack{P\in\Pi_{ab} \\ x=\text{length}(P)}} \{\pi(P)\}.$$

One can construct a fuzzy path from a fuzzy set $S = \{P_1, P_2, \ldots, P_n\}$ of crisp paths with the same tail vertex v_a and head vertex v_b, where $\pi(P)$ is the measure for each path $P \in S$. The fuzzy length of such a fuzzy path can be written in terms of the lengths $\ell_i = \text{length}(P_i)$ of the various paths P_i:

(43) $$\ell = \ell_1 \backslash \pi(P_1) + \ell_2 \backslash \pi(P_2) + \cdots + \ell_n \backslash \pi(P_n) .$$

2. Fuzzy Cuts

We define a *fuzzy cut* with a source vertex v_a and a sink vertex v_b on a graph G to be a Type II fuzzy graph with edge memberships $\mu(i)$ such that

(44) $$\mu(i) > 0 \text{ implies } \exists K \in \mathrm{K}_{ab} \text{ such that } \mu(j) \geq \mu(i), \forall e_i \in K$$

and

(45) $$\exists K \in \mathrm{K}_{ab} \text{ such that } \mu(j) = 1, \forall e_j \in K ,$$

where K_{ab} is the set of all crisp cuts between v_a and v_b. Equations (44) and (45) induce a membership for cuts $K \in \mathrm{K}_{ab}$ in the fuzzy cut:

(46) $$\kappa(K) = \min_{e_i \in K}\{\mu(i)\}.$$

Again, as a result of the normalization condition, there will always be at least one *most likely cut* between the source vertex and the sink vertex with $\mu(i) = 1$, $\forall e_i \in K$. If the graph is weighted, then the *fuzzy value* of the cut is defined as fuzzy number with membership

(47) $$\lambda(x) = \max_{\substack{K \in \mathrm{K}_{ab} \\ x = \mathrm{val}(K)}} \left\{ \min_{e_i \in K}\{\mu(i)\} \right\} = \max_{\substack{K \in \mathrm{K}_{ab} \\ x = \mathrm{val}(K)}} \{\kappa(K)\}.$$

One can construct a fuzzy cut from a fuzzy set $S = \{K_1, K_2, \ldots, K_n\}$ of crisp cuts between the source vertex v_a and a sink vertex v_b, where $\kappa(K)$ is the measure for each cut $K \in \mathrm{K}_{ab}$. The fuzzy value of such a fuzzy cut can be written in terms of the values $k_i = \mathrm{val}(K_i)$ of the various cuts K_i:

(48) $$k = k_1 \setminus \kappa(K_1) + k_2 \setminus \kappa(K_2) + \cdots + k_n \setminus \kappa(K_n).$$

3. Fuzzy Flows

We define a *fuzzy flow* with a source vertex v_a and a sink vertex v_b on a graph G, with edge weights w_i, to be a Type V fuzzy graph with edge weight memberships $\mu_i(x)$ such that

(49) $$\mu_i(x) > 0 \text{ implies } \exists F \in \Phi_{ab} \text{ such that } \mu_j\big(F(e_j)\big) \geq \mu_i\big(F(e_i)\big), \forall e_j \in G,$$

(50) $$\mu_i(x) \leq \hat{\mu}_{x \leq w_i},$$

and

(51) $$\exists F \in \Phi_{ab} \text{ such that } \mu_j\big(F(e_j)\big) = 1, \forall e_j \in G,$$

where Φ_{ab} is the set of all crisp flows between v_a and v_b and where $\hat{\mu}_{f_j \leq w_j}$ is taken in the sense of Equation (14a). Equations (49) and (51) induce a membership for flows $F \in \Phi_{ab}$ on the fuzzy flow:

(52) $$\varphi(F) = \min_{e_i \in G}\left\{\mu_i\big(F(e_i)\big)\right\},$$

Because of the normalization condition, there will always be at least one *most likely flow* for which $\mu_i(x) = 1$. The *fuzzy value* of the flow is defined as fuzzy number with membership

(53) $$\lambda(x) = \max_{\substack{F \in \Phi_{ab} \\ x = \mathrm{val}(F)}} \left\{ \min_{e_i \in G} \left\{ \mu_i \left(F(e_i) \right) \right\} \right\} = \max_{\substack{F \in \Phi_{ab} \\ x = \mathrm{val}(F)}} \left\{ \varphi(F) \right\}.$$

One can construct a fuzzy flow from a fuzzy set $S = \{F_1, F_2, \ldots F_n\}$ of crisp flows between the source vertex v_a and a sink vertex v_b, where $\varphi(F)$ is the measure for each flow $F \in \Phi_{ab}$. The fuzzy value of such a fuzzy flow can be written in terms of the values $u_i = \mathrm{val}(F_i)$ of the various flows F_i:

(54) $$u = u_1 \backslash \varphi(F_1) + u_2 \backslash \varphi(F_2) + \cdots + u_n \backslash \varphi(F_n).$$

III. Shortest Path

A. Formulation

1. General Formulation

Consider a fuzzy graph G with pure Type V fuzziness. (Graphs with Type II fuzziness also can be considered by treating the graph as a Type V graph where the weight has the possibility of being infinite—i.e., $\mu_{w_i}(\infty) > 0$.) Let Π be the set of all paths from vertex v_a to vertex v_b and let the fuzzy length of a path be

(55) $$\ell_P = \mathrm{length}(P) = \sum_{e_k \in P} w_k \text{ where } P \in \Pi.$$

The *fuzzy set of shortest paths* is a fuzzy set S on Π with memberships π_S given by

(56) $$\pi_S(P) = \min_{Q \in \Pi} \{\hat{\mu}_{\ell_P \le \ell_Q}\} \text{ where } P \in \Pi.$$

The support consists of all of the paths which potentially could have the minimum length,

(57) $$\mathrm{supp}(S) = \left\{ P \in \Pi \,\middle|\, \hat{\mu}_{\ell_P \le \ell_Q} > 0, \forall Q \in \Pi \right\},$$

and the measure associated with a given path is just the certainty that it is shorter than all other paths.

The fuzzy set of shortest paths defined above can be collapsed into a *fuzzy shortest path*, where each edge e_i has a membership in the fuzzy set S':

(58) $$\mu_{S'}(i) = \max_{e_i \in P, P \in \Pi} \{\pi_S(P)\} \text{ for } i = 1, \ldots, n_E.$$

One can write this alternately as

(59) $$\mu_{S'}(i) = \max_{e_i \in P, P\in\Pi} \left\{ \min_{Q\in\Pi} \{ \hat{\mu}_{\ell_P \leq \ell_Q} \} \right\} \text{ for } i = 1, \ldots, n_E .$$

The fuzzy shortest path satisfies the definition of a fuzzy path that we introduced in Equations (39) and (40). Equation (43) defines the fuzzy length of the fuzzy shortest path. Note that this produces a doubly fuzzy set (i.e., a fuzzy set of fuzzy numbers).

2. Alternate Formulation in Terms of Level Sets

An alternate approach to the shortest path problem is to consider the level graphs. Let G be a graph with purely Type V fuzziness. In considering the level-set formulation, we will restrict ourselves to the case where each weight w_i is a fuzzy convex number. Thus each α -level cut set w_i^α will consist of two (not necessarily distinct, if $\alpha = 1$) elements $w_i^{\alpha^-}$ and $w_i^{\alpha^+}$, with $w_i^{\alpha^-} \leq w_i^{\alpha^+}$. Now let G^α be the set of crisp graphs with edge i having the weight $w_i^{\alpha^+}$ or $w_i^{\alpha^-}$. Hence, G^α consists of at most 2^{n_E} crisp graphs. For $\alpha = 0$ we have $w_i^{\alpha^+} = \sup(\operatorname{supp} w_i)$ and $w_i^{\alpha^-} = \inf(\operatorname{supp} w_i)$. Define the *fuzzy set of shortest paths* for the fuzzy graph G to be the fuzzy set Σ on Π with membership function

(60) $$\eta_\Sigma(P) = \max_{\alpha\in(0,1]} \left\{ \alpha \,\middle|\, P \in \Sigma^\alpha \right\},$$

where

(61) $$\Sigma^\alpha = \left\{ P \in \Pi \,\middle|\, P \text{ is a shortest path of some graph in } G^\alpha \right\}.$$

Consequently, the support of the fuzzy set of shortest paths formulated via level sets is

(62) $$\operatorname{supp}(\Sigma) = \left\{ P \in \Pi \,\middle|\, P \in \Sigma^\alpha \text{ for some } \alpha \in (0,1] \right\}.$$

It is also possible to collapse the fuzzy set of shortest paths in this formulation into a fuzzy shortest path satisfying Equations (39) and (40), where each edge e_i has a membership in the fuzzy set Σ':

(63) $$\mu_{\Sigma'}(i) = \max_{e_i \in P, P\in\Pi} \{ \eta_\Sigma(P) \} \text{ for } i = 1, \ldots, n_E .$$

3. Relationship between the Two Formulations

A natural question to ask is what is the relationship between the fuzzy set of shortest paths as defined in general, Equation (56), and that defined via the level-set formulation, Equation (60). The following is a partial answer to this question.

Claim: Let G be a graph with Type V fuzziness and with edge weights w_i, which are convex fuzzy numbers. Then, $\operatorname{supp}(\Sigma) \subseteq \operatorname{supp}(S)$. This inclusion is proper in general.

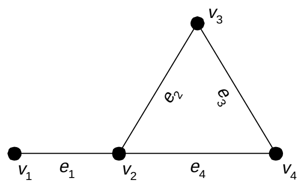
Proof:

Part (A): Let $P \in \operatorname{supp}(\Sigma)$, so by definition $P \in \Sigma^{\alpha}$ for some $\alpha > 0$. Since the edge weights are continuous, we may assume $0 < \alpha < 1$. Define $\kappa = \min\left\{\sup_{Q\in\Pi}\left\{\operatorname{supp}(\ell_Q)\right\}\right\}$ and let $Q_\kappa \in \Pi$ be a path such that $\kappa \equiv \sup\left\{\operatorname{supp}(\ell_{Q_\kappa})\right\}$. Notice that

$$\sum_{e_i\in P} \inf\left\{\operatorname{supp}(w_i)\right\} \le \sum_{e_i\in P} w_i^{\alpha^-} < \sum_{e_i\in P} w_i^{\alpha^+} \le \sum_{e_j\in Q_\kappa} w_j^{\alpha^+} \le \sum_{e_j\in Q_\kappa} \sup\left\{\operatorname{supp}(w_j)\right\} = \kappa .$$

Here the strict inequality follows from $\alpha < 1$. Hence by Equation (64)—which will be proved in the next section—we have $P \in \operatorname{supp}(S)$. Therefore, $\operatorname{supp}(\Sigma) \subseteq \operatorname{supp}(S)$.

Part (B): Here we present an example where $\operatorname{supp}(S) \not\subset \operatorname{supp}(\Sigma)$. Let G be the following graph:



Let the edge weights be triangular fuzzy numbers (see the Appendix) $w_1 = w_4 = [0,1,2]$ and $w_2 = w_3 = [1,2,4]$. There are two paths from v_1 to v_4, $P=(e_1, e_2, e_3)$ and $Q=(e_1, e_4)$, with lengths $\ell_P = [2,5,10]$ and $\ell_Q = [0,2,4]$. Thus $\kappa = 4$, and Q is the path where this is obtained. Since $\hat{\mu}_{\ell_P \le \ell_Q} > 0$, we have $P \in \operatorname{supp}(S)$. However, we claim that $P \notin \Sigma^{\alpha}$ for any α.

The α -level cut sets of the edge weights for $\alpha \in (0,1]$ are $w_1^{\alpha} = w_4^{\alpha} = \{\alpha, 2-\alpha\}$ and $w_2^{\alpha} = w_3^{\alpha} = \{1+\alpha, 4-2\alpha\}$. Thus, there are eight graphs in the set G^{α} for $\alpha \in (0,1]$. For any graph $g \in G^{\alpha}$, notice that $\ell_P^g = w_1^{\alpha^g} + w_2^{\alpha^g} + w_3^{\alpha^g}$ and $\ell_Q^g = w_1^{\alpha^g} + w_4^{\alpha^g}$. Thus, for any level graph G^{α}, we have $\ell_P^g \le \ell_Q^g$ if and only if $w_2^{\alpha^g} + w_3^{\alpha^g} \le w_4^{\alpha^g}$. However, the maximum value for w_4^{α} is $2-\alpha$ whereas the minimum value for $w_2^{\alpha^g} + w_3^{\alpha^g}$ is the sum of their respective minimum values, i.e., $2+2\alpha$. Since $2+2\alpha > 2-\alpha$ for $\alpha \in (0,1]$, we have $w_2^{\alpha^g} + w_3^{\alpha^g} > w_4^{\alpha^g}$. Hence, $P \notin \Sigma^{\alpha}$ for any $\alpha \in (0,1]$, and thence $P \notin \operatorname{supp}(\Sigma)$.

B. Algorithm

We said previously that the set of possible shortest paths is

(57) $$\operatorname{supp}(S) = \left\{P \in \Pi \,\middle|\, \hat{\mu}_{\ell_P \le \ell_Q} > 0, \forall Q \in \Pi\right\}.$$

In order to solve the fuzzy shortest path problem algorithmically, it is useful to consider the following estimation:

(64) $$\left\{P \in \Pi \,\middle|\, \inf\left\{\operatorname{supp}(\ell_P)\right\} < \kappa\right\} \subseteq \operatorname{supp}(S) \subseteq \left\{P \in \Pi \,\middle|\, \inf\left\{\operatorname{supp}(\ell_P)\right\} \le \kappa\right\}$$

where

(65) $$\kappa = \min_{P\in\Pi}\left\{\sup\left\{\mathrm{supp}(\ell_P)\right\}\right\}.$$

Proof:

Case (A): Let $P\in\Pi$ such that $\hat{\mu}_{\ell_P\le\ell_Q} > 0$ for $\forall Q\in\Pi$. Since there are only a finite number of paths $Q\in\Pi$, κ is the minimum of a finite set of numbers. Hence there exists $Q\in\Pi$ such that $\kappa = \sup\left\{\mathrm{supp}(\ell_Q)\right\}$. By hypothesis $\hat{\mu}_{\ell_P\le\ell_{Q_\kappa}} > 0$, hence, $\exists p\in\mathrm{supp}(\ell_P)$ and $\exists q\in\mathrm{supp}(\ell_Q)$ such that $p\le q\le\sup\left\{\mathrm{supp}(\ell_Q)\right\}=\kappa$. Thus, $\inf\left\{\mathrm{supp}(\ell_P)\right\}\le p\le q\le\kappa$, so $\inf\left\{\mathrm{supp}(\ell_P)\right\}\le\kappa$.

Case (B): Let $P\in\Pi$ such that $\inf\left\{\mathrm{supp}(\ell_P)\right\}<\kappa$. Let $p\in\mathrm{supp}(\ell_P)$ with $p<\kappa$, and let $Q\in\Pi$. Since $\kappa\le\sup\left\{\mathrm{supp}(\ell_Q)\right\}$, there is some $q\in\mathrm{supp}(\ell_Q)$ such that $q>p$. Hence, $\hat{\mu}_{\ell_P\le\ell_Q}>0$. Since Q was arbitrary, $P\in\mathrm{supp}(S)$.

Remark: If $\inf\left\{\mathrm{supp}(\ell_P)\right\}=\kappa$, then $P\in\mathrm{supp}(S)$ if and only if $\mu_{\ell_P}(\kappa)>0$ and for $\forall Q\in\Pi$ such that $\inf\left\{\mathrm{supp}(\ell_Q)\right\}=\kappa$ we have $\mu_{\ell_Q}(\kappa)>0$.

Now consider the graph $\underline{G}$ that is identical to G, except the edge weights are crisp: $\underline{w}_i=\inf\left\{\mathrm{supp}(w_i)\right\}$. Hence we have an isomorphism between the set of paths Π in G and the set of paths $\underline{\Pi}$ in $\underline{G}$. Let $P\in\Pi$ and $\underline{P}\in\underline{\Pi}$ be corresponding paths in the two graphs: $E_P=\left\{e_{P_1},e_{P_2},\ldots,e_{P_n}\right\}$ and $E_{\underline{P}}=\left\{e_{\underline{P}_1},e_{\underline{P}_2},\ldots,e_{\underline{P}_n}\right\}$ represent the set of edges that occur in P and $\underline{P}$, respectively; analogously, w_{P_i} and $\underline{w}_{\underline{P}_i}$ will represent the edge weights, so we can write $\ell_P=\sum_{i=1}^{n}w_{P_i}$ and $\ell_{\underline{P}}=\sum_{i=1}^{n}\underline{w}_{\underline{P}_i}$.

Claim: $\inf\left\{\mathrm{supp}(\ell_P)\right\}=\inf\left\{\mathrm{supp}\left(\sum_{i=1}^{n}w_{P_i}\right)\right\}=\sum_{i=1}^{n}\inf\left\{\mathrm{supp}(w_{P_i})\right\}=\sum_{i=1}^{n}\underline{w}_{\underline{P}_i}=\ell_{\underline{P}}$

Proof:

The first, third, and fourth equalities are a direct application of the definitions of ℓ_P, $\underline{w}_{\underline{P}_i}$ and $\ell_{\underline{P}}$ respectively. The second equality follows almost directly from the definition of fuzzy addition. Using the α-level cut definition of addition, we see that $\left(\sum_{i=1}^{n}w_{P_i}\right)^\alpha=\sum_{i=1}^{n}\left(w_{P_i}\right)^\alpha$ and $\left(\sum_{i=1}^{n}w_{P_i}\right)_\alpha=\sum_{i=1}^{n}\left(w_{P_i}\right)_\alpha$ for all $\alpha>0$. Hence, in particular, $\inf\left(\sum_{i=1}^{n}w_{P_i}\right)_\alpha=\inf\sum_{i=1}^{n}\left(w_{P_i}\right)_\alpha$. Since this holds for arbitrary $\alpha>0$, we can take limits on both sides to obtain

$$\inf\left\{\mathrm{supp}\left(\sum_{i=1}^{n}w_{P_i}\right)\right\}=\lim_{\alpha\to0}\inf\left(\sum_{i=1}^{n}w_{P_i}\right)_\alpha=\lim_{\alpha\to0}\inf\sum_{i=1}^{n}\left(w_{P_i}\right)_\alpha=\sum_{i=1}^{n}\inf\left\{\mathrm{supp}\left(w_{P_i}\right)\right\},$$

which substantiates the claim.

Similarly, we consider the graph $\overline{G}$ that is identical to G, except the edge weights are crisp: $\overline{w}_i = \sup\{\mathrm{supp}(w_i)\}$. Once again, the paths $\overline{P} \in \overline{\Pi}$ in $\overline{G}$ correspond to $P \in \Pi$ in G, where the lengths are given by $\ell_{\overline{P}} = \sum_{i=1}^{n} w_{\overline{P}_i}$.

Claim: $\sup\{\mathrm{supp}(\ell_P)\} = \sup\left\{\mathrm{supp}\left(\sum_{i=1}^{n} w_{P_i}\right)\right\} = \sum_{i=1}^{n} \sup\{\mathrm{supp}(w_{P_i})\} = \sum_{i=1}^{n} \overline{w}_{\overline{P}_i} = \ell_{\overline{P}}$

Proof:
The proof of this claim is analogous to that of the previous claim.

We can now re-express Equations (64) and (65) as

(66) $$\left\{P \in \Pi \,\middle|\, \ell_{\underline{P}} < \kappa\right\} \subseteq \mathrm{supp}(S) \subseteq \left\{P \in \Pi \,\middle|\, \ell_{\underline{P}} \le \kappa\right\},$$

where

(67) $$\kappa = \min_{P \in \Pi}\left\{\ell_{\overline{P}}\right\}.$$

Thus we have reduced the fuzzy shortest path problem to a pair of crisp shortest path problems: (i) find κ, the length of the shortest path in $\overline{G}$; and (ii) find S, the paths with lengths less than κ in $\underline{G}$. Special consideration should be given to boundary points. Numerous methods are available for solving the first problem; the second problem can be solved by adapting algorithms for the *k*-shortest path problem [Sh 79; To 88; Ye 71].

C. *NP Completeness*

Given a graph G and two fixed vertices v_a and v_b, the longest path problem "Does there exist a path from v_a to v_b of length greater than or equal to κ?" is NP-complete in general [GJ 79]. Thus, for a crisp graph with crisp weights $\underline{w}_i = \inf\{\mathrm{supp}(w_i)\}$, finding all paths of length less than κ is also an NP-complete problem. Hence, any algorithm for computing all paths of length less than κ is NP-hard, as is the fuzzy shortest path problem. Note that the longest path problem can be solved in polynomial time for directed acyclic graphs [La 76].

D. *Example*

Next we solve a simple fuzzy shortest path problem for the Type V fuzzy digraph shown in Figure 1. The fuzzy lengths for the four paths from vertex *a* to vertex *f* are listed in Figure 2—from this we see that $\kappa = 8$ and that path *abdf* has membership $\pi_S(abdf) = 1$, path *abef* has membership $\pi_S(abef) = 2/5$, and the other paths have membership $\pi_S(acdf) = \pi_S(acef) = 0$ in the fuzzy set of shortest paths. Figure 3 illustrates the fuzzy shortest path.

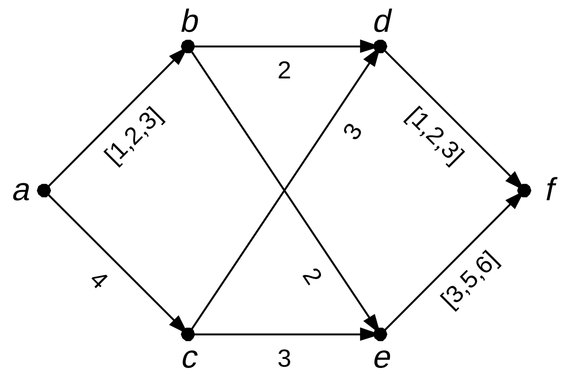


Figure 1. Example Type V fuzzy directed graph. The vertex *a* is the source/origin and the vertex *f* is the sink/destination. The edge weights are either crisp numbers or fuzzy triangular numbers (see Appendix).

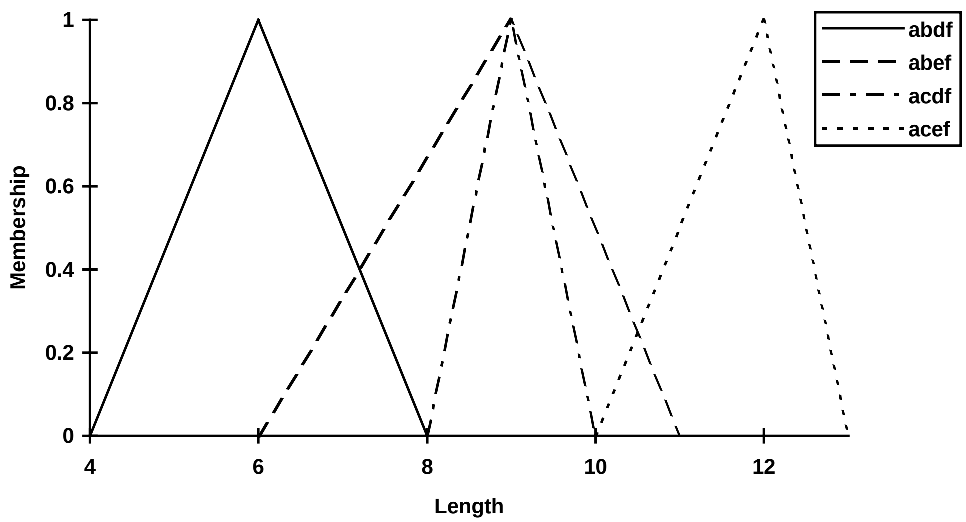


Figure 2. Fuzzy path lengths for the paths from vertex *a* to vertex *f* of the graph in Figure 1.

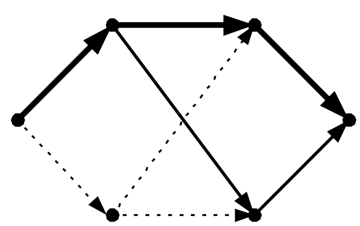


Figure 3. The fuzzy shortest path from vertex *a* to vertex *f* of the graph in Figure 1. The thick solid lines have membership 1, the thin solid lines have membership 2/5, and the dotted lines have membership 0.

As a more practical example of the theory, we consider the problem of finding the fuzzy shortest path (in terms of travel time) from Santa Fe, New Mexico, to Monticello, Utah, in the United States. Figure 4 shows the vertices and edges in our representation of the highway network between the two cities. Each edge has been assigned a fuzzy travel time based on the length of the highway segment as well as a fuzzy travel speed along it. We use the triangular representation (see the Appendix) for the fuzzy numbers involved.

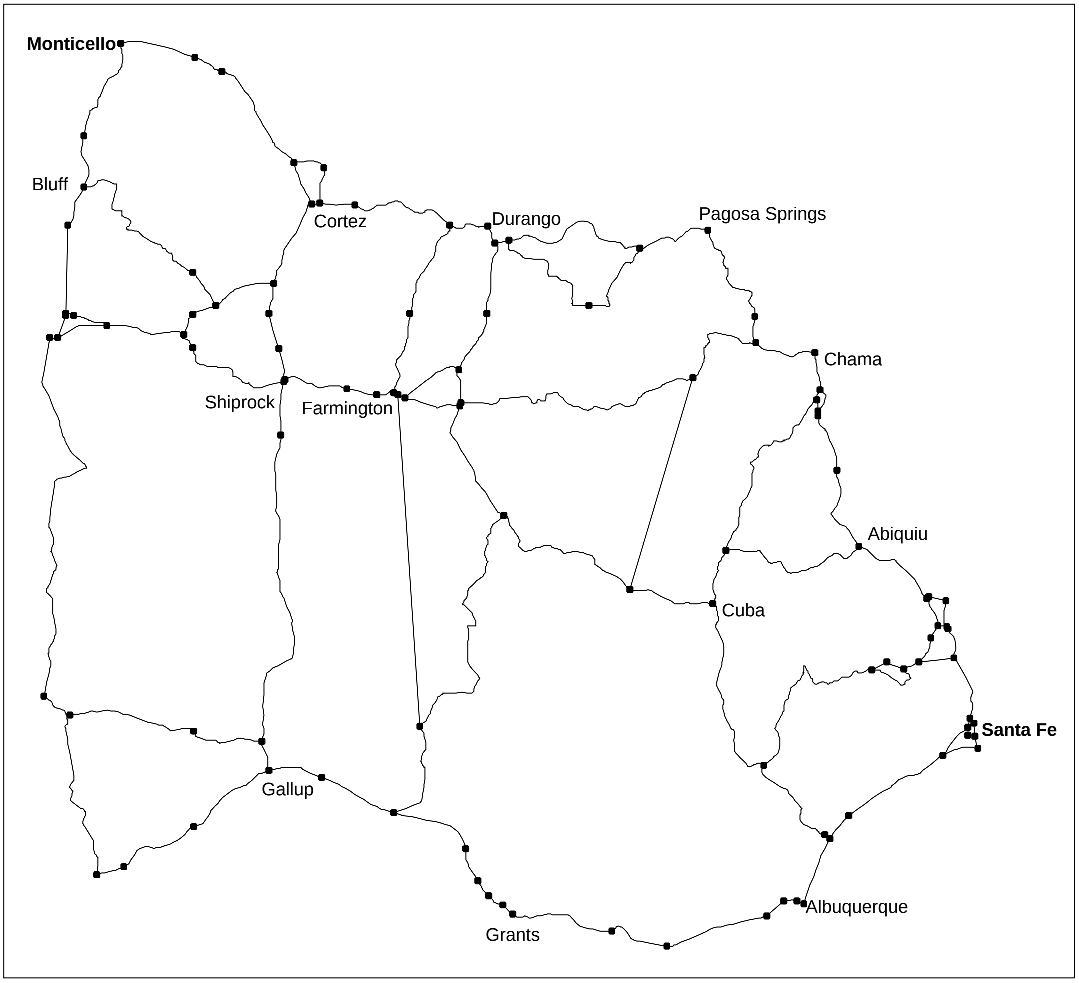


Figure 4. Highway map of the four-corners area of the United States. In this example, we want to find the path fuzzy shortest path (in terms of travel time) from Santa Fe to Monticello.

By applying the methods of the preceding section, we have identified the fifteen paths in the support of the fuzzy set of shortest paths. Figure 5 and Figure 6 show the fuzzy lengths (i.e., travel times) and memberships for the paths in the support. Figure 7 presents the six paths with the greatest membership in the fuzzy set. If we use Equation (59) to collapse the fuzzy set of shortest paths into a fuzzy shortest path, we arrive at the graph in Figure 8 that solves the problem posed here.

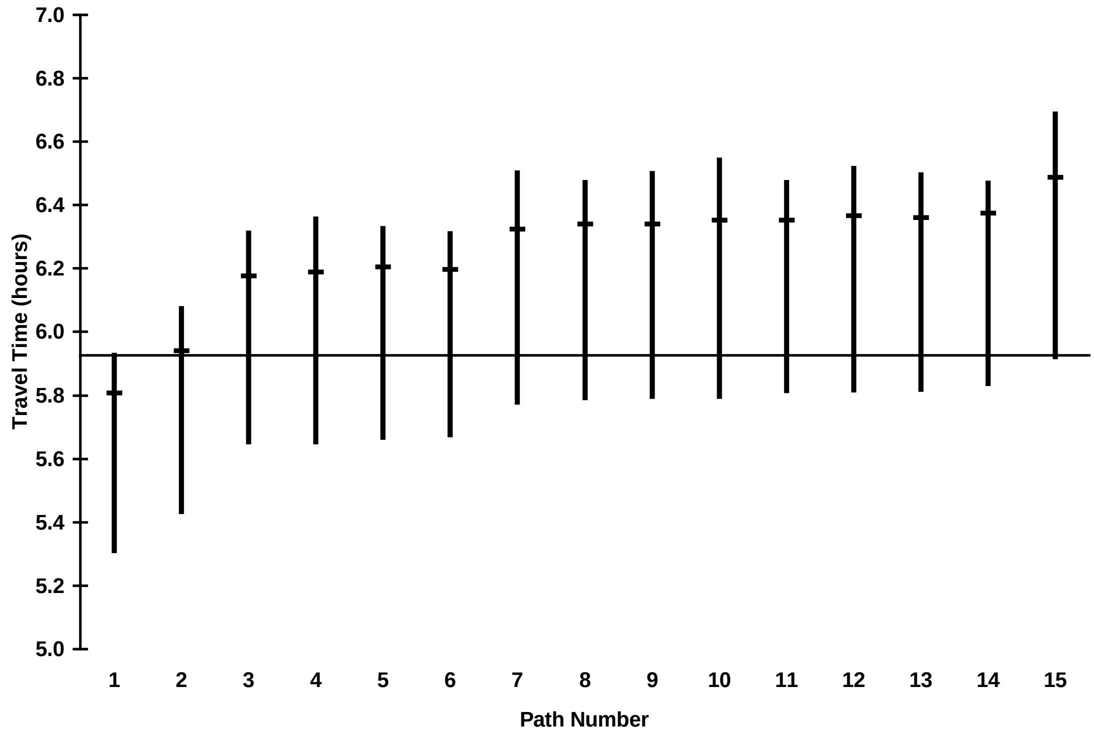


Figure 5. Fuzzy path lengths for the paths in the support of the fuzzy set of shortest paths for the problem in Figure 4. The horizontal line shows the value of κ given by Equation (65).

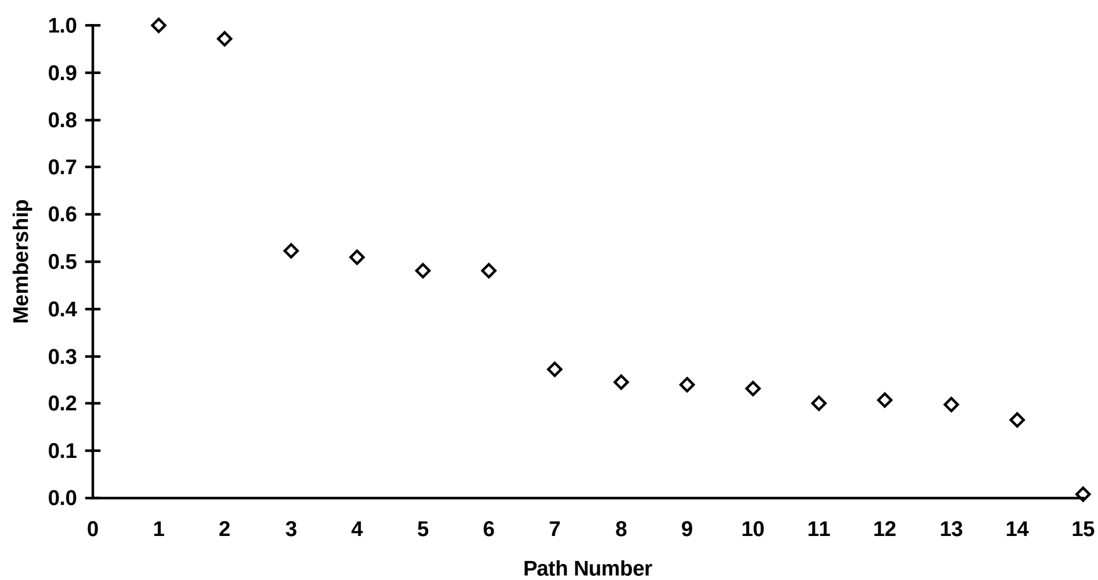


Figure 6. Memberships of the paths in the support of the fuzzy set of shortest paths for the problem in Figure 4.

Figure 7. The six paths (dark lines) with the greatest membership in the fuzzy set of shortest paths for the problem in Figure 4.

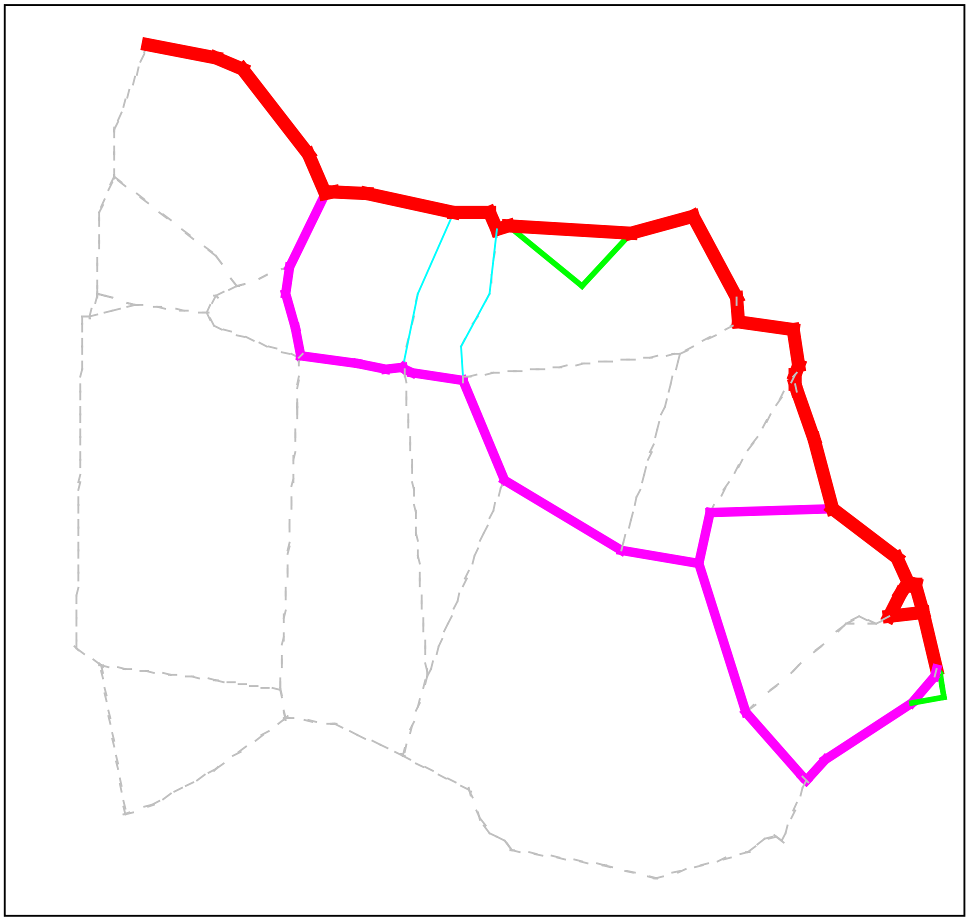


Figure 8. The fuzzy shortest path for the problem in Figure 4. The thickness of the edge is proportional to its membership in the path. The solid line thickness represents the membership range: thickest for $\mu \in$ (¾, 1], thick for $\mu \in$ (½, ¾], thin for $\mu \in$ (¼, ½], and thinnest for $\mu \in$ (0, ¼]. The dashed lines have membership zero.

IV. Minimum Cut

A. Formulation

1. General Formulation

Consider a fuzzy graph G with pure Type V fuzziness. (Graphs with Type II fuzziness also can be considered by treating the graph as a Type V graph where the weight has the possibility of being zero—i.e., $\mu_{w_i}(0) > 0$.) Let K be the set of all proper cuts with a source vertex v_a and a sink vertex v_b, and let the value of a cut be

(68) $$k_K = \mathrm{val}(K) = \sum_{e_k \in \mathrm{forward}(K)} w_k \quad \text{where } K \in \mathrm{K}.$$

Note that if only proper cuts are considered, all of the edges in K will be forward. The *fuzzy set of minimum cuts* is a fuzzy set S on K with memberships

(69) $$\kappa_S(K) = \min_{L \in \mathrm{K}} \left\{ \hat{\mu}_{k_K \le k_L} \right\} \text{ where } K \in \mathrm{K} .$$

The support consists of all the cuts which could have the minimum capacity,

(70) $$\operatorname{supp}(S) = \left\{ K \in \mathrm{K} \,\middle|\, \hat{\mu}_{k_K \le k_L} > 0, \forall L \in \mathrm{K} \right\},$$

and the measure associated with a given cut is the certainty that its value is less than the value of any other cut.

As with the shortest path, this fuzzy set of minimum cuts can be collapsed into a *fuzzy minimum cut*, where each edge e_i has a membership

(71) $$\mu_{S'}(i) = \max_{e_i \in K, K \in \mathrm{K}} \left\{ \kappa_S(K) \right\} \text{ for } i = 1, \ldots, n_E$$

in the fuzzy set S'. This can also be written as

(72) $$\mu_{S'}(i) = \max_{e_i \in K, K \in \mathrm{K}} \left\{ \min_{L \in \mathrm{K}} \left\{ \hat{\mu}_{k_K \le k_L} \right\} \right\} \text{ for } i = 1, \ldots, n_E .$$

Note that this satisfies the definition of fuzzy cut presented in Equations (44) and (45). Equation (48) defines the fuzzy value of the fuzzy minimum cut.

2. Alternate Formulation in Terms of Level Sets

We will again consider an alternate formulation in terms of level sets. As in the case for the shortest path problem, let G^α be the set of crisp graphs with edge weight $w_i^{\alpha^+}$ or $w_i^{\alpha^-}$. Recall that for convex weights G^α consists of 2^{n_E} graphs for $\alpha \in (0,1)$. We define the *fuzzy set of minimum cuts* for the fuzzy graph G to be the fuzzy set Σ on K with membership function

(73) $$\eta_\Sigma(K) = \max_{\alpha \in [0,1]} \left\{ \alpha \,\middle|\, K \in \Sigma^\alpha \right\}$$

where

(74) $$\Sigma^\alpha = \left\{ K \in \mathrm{K} \,\middle|\, K \text{ is a minimum cut of some graph in } G^\alpha \right\}.$$

Thus we find that the support of the fuzzy set of minimum cuts is

(75) $$\operatorname{supp}(\Sigma) = \left\{ K \in \mathrm{K} \,\middle|\, K \in \Sigma^\alpha \text{ for some } \alpha \in (0,1] \right\}.$$

Again it is possible to collapse this fuzzy set of minimum cuts into a fuzzy minimum cut where each edge e_i has membership

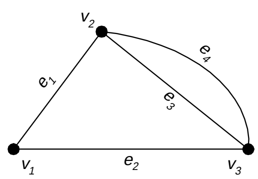
(76) $$\mu_{\Sigma'}(i) = \max_{e_i \in K, K \in \mathrm{K}} \left\{ \eta_\Sigma(K) \right\} \text{ for } i = 1, \ldots, n_E$$

in the fuzzy set Σ'.

3. Relationship between the Two Formulations

As with shortest path, we find that the level-set formulation gives a smaller set than the general formulation: namely, $\operatorname{supp}(\Sigma) \subseteq \operatorname{supp}(S)$. Again, the inclusion is proper in general.

Proof:
The proof is a straightforward generalization of the corresponding proof for the shortest path. For Part (A), simply replace all paths with cuts and lengths with cut capacities. For part (B), we use a similar graph for a counter-example. Let G' be the following graph with source at v_1 and sink at v_3:



Choose $w_1 = [2,3,4]$, $w_2 = [1,2,3]$, and $w_3 = w_4 = [2,3,4]$. In this case, let $L = \{e_1, e_2\}$ and $K = \{e_2, e_3, e_4\}$. Inspection yields $\kappa = 7$ with L as the cut when this is obtained. Since $\hat{\mu}_{k_K \le k_L} > 0$, we have $K \in S$. Again, by using the same technique with level cuts as in the proof in the shortest path section, one can easily obtain $K \notin \Sigma^{\alpha}$ for any $\alpha \in (0,1]$.

B. Algorithm

In analogy with how we proceeded in the shortest path problem, we can reformulate Equation (70) to estimate the possible minimum cuts as

(77) $$\left\{K \in \mathcal{K} \,\middle|\, \min\{\operatorname{supp}(k_K)\} < \kappa\right\} \subseteq \operatorname{supp}(S) \subseteq \left\{K \in \mathcal{K} \,\middle|\, \min\{\operatorname{supp}(k_K)\} \le \kappa\right\},$$

where

(78) $$\kappa = \min_{K \in \mathcal{K}} \left\{\sup\left\{\operatorname{supp}(k_K)\right\}\right\}.$$

(We omit the proofs in this section, as they are analogous to the corresponding ones for shortest paths.) Once again, we consider the graphs $\underline{G}$ and $\overline{G}$ that are identical to G, except that the edge weights are crisp: $\underline{w}_i = \inf\{\operatorname{supp}(w_i)\}$ and $\overline{w}_i = \inf\{\operatorname{supp}(w_i)\}$, respectively. This permits us to re-express Equations (77) and (78) as

(79) $$\left\{K \in \mathcal{K} \,\middle|\, k_{\underline{K}} < \kappa\right\} \subseteq \operatorname{supp}(S) \subseteq \left\{K \in \mathcal{K} \,\middle|\, k_{\underline{K}} \le \kappa\right\},$$

where

(80) $$\kappa = \min_{K \in \mathcal{K}} \left\{k_{\overline{K}}\right\}.$$

Thus we have reduced the fuzzy minimum cut problem to a pair of crisp minimum cut problems: (i) find κ, the value of the minimum cut in $\overline{G}$; and (ii) find S, the cuts with values less than κ in $\underline{G}$. Again, the boundary values need to be checked separately. Numerous methods are available for solving the first problem. The second problem can be dealt with via the following recursive procedure: Solve the minimum cut problem on $g := \underline{G}$ to find a minimum cut $\{e_1, e_2, \ldots, e_n\}$ containing n edges. Now consider graphs $g_1, g_2, \ldots, g_n$ that differ from g in that the head and tail vertices of edge e_i are merged into a single vertex: i.e., construct g_i by removing the vertex t_i from g and by making any edges connected to t_i in g connect to h_i in g_i. Next solve the minimum cut problems on $g_1, g_2, \ldots, g_n$. If the value of the minimum cut for g_i is less than or equal to κ, then the cut belongs to S and the foregoing procedure must be repeated with $g := g_i$. This algorithm was inspired by a graph contraction algorithm for enumerating all minimum cuts [MR 96].

C. NP Completeness

Given a graph G, the maximum cut problem "Does there exist a separation of the vertices in G into two disjoint subsets V_1 and V_2 such that the sum of the weights of edges with tail in V_1 and head in V_2 is greater than or equal to κ?" is NP-complete in general [GJ 79]. Thus, for a crisp graph with crisp weights $\underline{w}_i = \inf\{\mathrm{supp}(w_i)\}$, finding all cuts of capacity less than κ is also an NP-complete problem. Thus, the general fuzzy minimum cut problem is NP-hard.

D. Example

We now solve the fuzzy minimum cut problem for the example digraph in Figure 9, considering the cuts that separate the source vertex *a* from the sink vertex *f*. Figure 10 shows the values of the cuts in the support of the fuzzy set of minimum cuts, and Figure 11 shows the cuts themselves. The three cuts can be collapsed into the fuzzy minimum cut shown in Figure 12.

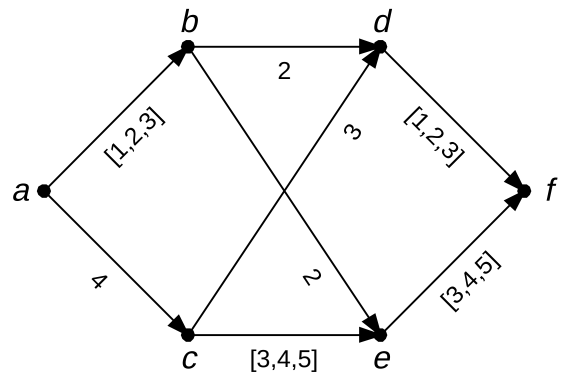


Figure 9. Example directed graph for the fuzzy minimum cut problem. We consider cuts that separate the source vertex *a* from the sink vertex *f*. The edge weights are either crisp numbers or fuzzy triangular numbers (see Appendix).

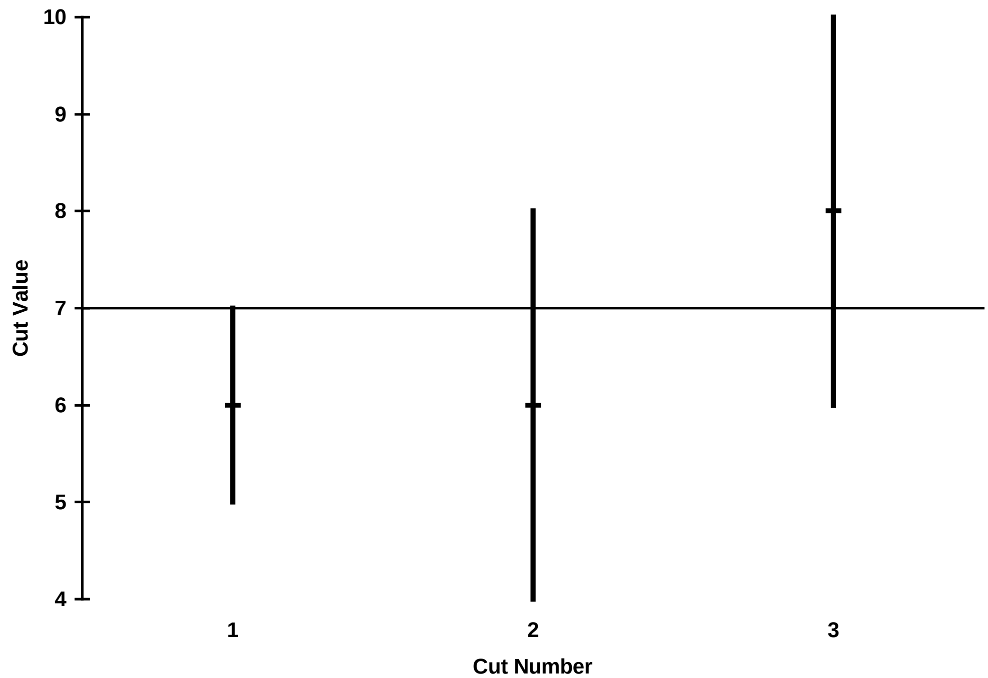


Figure 10. Fuzzy minimum cut values for the cuts in the support of the fuzzy set of minimum cuts for the problem in Figure 9. The horizontal line shows the value of κ given by Equation (78).

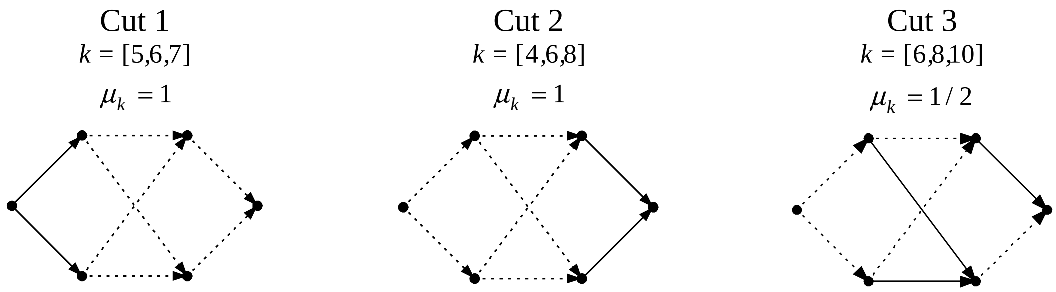


Figure 11. The cuts (solid lines) in the support of the fuzzy set of minimum cuts for the problem in Figure 9.

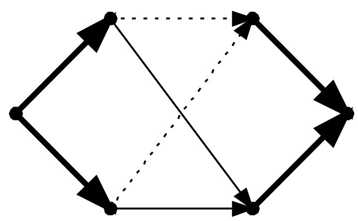


Figure 12. The fuzzy minimum cut for the problem in Figure 9. The thickness of the edge is corresponds to its membership in the cut: the thick lines represent μ = 1, the thin ones represent μ = ½, and the dotted lines have membership 0.

V. Maximum Flow

A. Formulation

Consider a fuzzy graph G with pure Type V fuzziness. (Graphs with Type II fuzziness also can be considered by treating the graph as a Type V graph where the weight has the possibility of being zero—i.e., $\mu_{w_i}(0) > 0$.) Let Φ be the set of all crisp flows on the graph $\overline{G}$ with a source vertex v_a and a sink vertex v_b and let the value of a flow be

(81) $$u_F = \text{val}(F) = \sum_{\substack{j=1,\dots,n_E \\ t_j = v_a}} f_j = \sum_{\substack{j=1,\dots,n_E \\ h_j = v_b}} f_j ,$$

where $f_i \equiv F(e_i)$. We define the degree to which a flow satisfies the fuzzy edge weights as

(82) $$\gamma(F) = \min_{e_i \in G} \left\{ \mu_{f_i \le w_i} \right\}.$$

The *fuzzy set of maximum flows* is a fuzzy set S on Φ with memberships

(83) $$\varphi_S(F) = \begin{cases} \gamma(F) & \text{if } u_F \ge u_{\max} \\ 0 & \text{otherwise} \end{cases} \quad \text{where } F \in \Phi$$

and where

(84) $$u_{\max} = \max_{\gamma(H)=1, H\in\Phi} \{u_H\}$$

is the largest value of the flow with unit membership. The support consists of all the flows which could have the maximum flow,

(85) $$\operatorname{supp}(S) = \left\{ F \in \Phi \,\middle|\, \gamma(F) > 0, u_F \geq u_{\max} \right\},$$

and the measure associated with a given flow is the certainty that its value is greater than the value of any other flow.

As with the shortest path, the fuzzy set of maximum flows can be collapsed into a *fuzzy maximum flow*, where each edge flow f_i has a membership in the fuzzy set S':

(86) $$\mu_{S_i'}(x) = \max_{f_i = x, F\in\Phi} \{\varphi_S(F)\} \text{ for } i = 1, \ldots, n_E .$$

This can also be written as

(87) $$\mu_{S_i'}(x) = \max_{f_i = x, u_F \geq u_{\max}, F\in\Phi} \{\gamma(F)\} \text{ for } i = 1, \ldots, n_E .$$

Note that this satisfies the definition of fuzzy flow presented in Equations (49), (50), and (51). Equation (54) defines the fuzzy value of the fuzzy maximum flow.

B. Algorithm

Solving the fuzzy maximum flow problem stated above requires enumerating all of the flows in $\overline{G}$ that have a value greater than $u_{\max}$—a difficult problem even for situations where the flows are integers. It is possible, however, to find an upper bound on the value of the flow by solving the maximum flow problem for the crisp graph $\overline{G}$. This provides us with the support of the fuzzy maximum flow value:

(88) $$\left[u_{\max}, \sup_{F\in\underline{G}} \{u_F\} \right].$$

There may also be a relationship between the fuzzy maximum flow value and the fuzzy minimum cut value, as there is for crisp graphs.

C. Example

As an example, we consider the fuzzy maximum flow problem from vertex *a* to vertex *f* of the graph in Figure 13. This problem is simple enough to solve by the exhaustive application of Equation (87); Figure 14 illustrates the fuzzy maximum flow. The value of the fuzzy maximum flow is $[6,6,7]$.

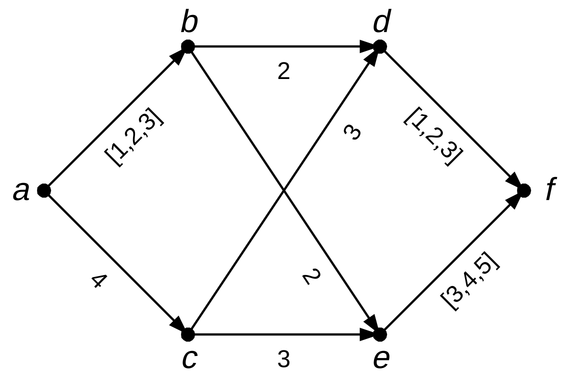


Figure 13. Example directed graph for the fuzzy minimum cut problem. We consider cuts that separate the source vertex *a* from the sink vertex *f*. The edge weights are either crisp numbers or fuzzy triangular numbers (see Appendix).

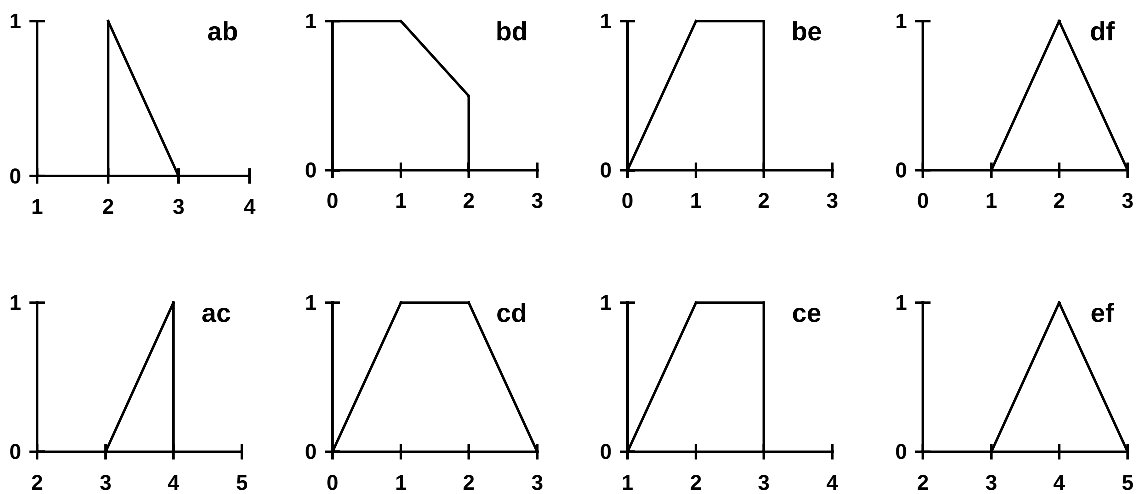


Figure 14. Membership functions for the flows on edges of the fuzzy maximum flow from vertex *a* to vertex *f* of the graph in Figure 9.

VI. Articulation Points

A. Formulation

For a crisp, connected, and undirected graph, an articulation point is a vertex that, when removed from the graph, makes the graph disconnected [Se 92]. When generalizing this concept to fuzzy graphs, it is most interesting to consider graphs with Type I' fuzziness. (Fuzzy graphs of Types II, III, and IV can be treated by expanding them to Type I' using Equation (38).) Let $\wp(V)$ be the power set for V (i.e., the set of all subsets of V). The *fuzzy set of articulation points* is a fuzzy set A on $\wp(V)$ with memberships λ_A given by

(89) $$\lambda_A(S) = \max_{j=1,..,n_G} \left\{ \min\left\{ \mu_j, \omega_{S,j} \right\} \right\} \text{ where } S \in \wp(V),$$

where $\omega_{S,j}$ is unity if S is the articulation points for the graph G_j, and zero otherwise. Note that this definition preserves the unit normalization condition, since there is always some j for which μ_j is unity and $\omega_{S,j}$ is also unity for some S (even if S is the empty set).

The fuzzy set of articulation points defined above can be collapsed into *fuzzy articulation points*, where each vertex v_i has a membership

(90) $$\lambda_{A'}(i) = \max_{v_i \in S}\left\{\lambda_A(S)\right\} \text{ for } i = 1,\ldots,n_V$$

in the fuzzy set A'. This is equivalent to assigning the membership via

(91) $$\lambda_{A'}(i) = \max_{j=1,..,n_G}\left\{\min\left\{\mu_j, \omega_{i,j}\right\}\right\} \text{ for } i = 1,\ldots,n_V\ ,$$

where $\omega_{i,j}$ is unity if v_i is an articulation point in the graph G_j, and zero otherwise. Note that unit normalization is preserved by this operation only if there exists a graph G_j that has unit normalization $\mu_j = 1$ and some articulation points:

(92) $$\exists j, \mu_j = 1 \text{ and } \omega_{S,j} = 1 \text{ and } S \neq \varnothing .$$

B. Algorithm

For graphs with Type I' fuzziness, one can apply any standard algorithm for finding articulation points on crisp graphs and use Equation (91) as the recipe for dealing with the fuzziness. In the case of Type II, III, or IV fuzzy graphs, there may be more efficient approaches than that of expanding the graph to Type I' and then proceeding with the algorithm just mentioned.

C. Example

Figure 15 presents an articulation point problem for a Type I' fuzzy graph. The articulation points for the component graphs are given in Table 1. Thus the fuzzy set of articulation points is $A = \{b,c\}\backslash 1 + \{b,e,f\}\backslash 0.75 + \varnothing\backslash 0.25$. This can be collapsed into the fuzzy articulation points $A' = b\backslash 1 + c\backslash 1 + e\backslash 0.75 + f\backslash 0.75$ shown in Figure 16.

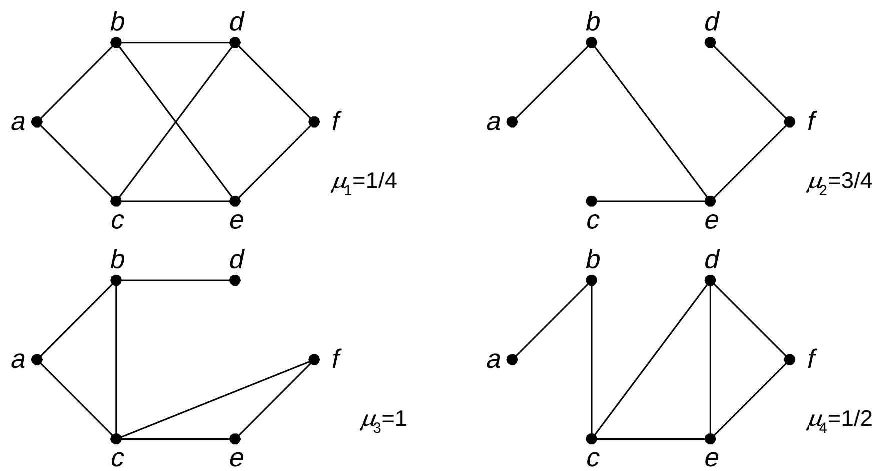


Figure 15. An example Type I' fuzzy graph for the articulation point problem.

Table 1. The articulation points for the graph in Figure 15.

i	μ_i	S_i
1	¼	$\varnothing$
2	¾	$\{b, e, f\}$
3	1	$\{b, c\}$

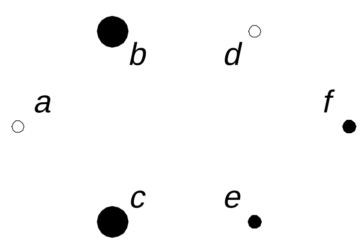


Figure 16. The fuzzy articulation points for the graph in Figure 15. The large, solid vertices have membership 1, the small, solid vertices have membership ¾, and the hollow vertices have membership 0.

VII. Conclusion

This paper has classified how fuzziness can be incorporated into classical graph theory and has emphasized the uniform application of a few key principles to the fuzzy graph problem: the construction of fuzzy graph membership grades via the ranking of fuzzy numbers, the preservation of membership grade normalization, and the collapse of fuzzy sets of graphs into fuzzy graphs play a unifying role in the handling of fuzziness in graphs. This allows for a coherent treatment of the classic shortest path, maximum flow, minimum cut, and articulation point problems. It also simplifies considerably the development of algorithms for the solution of these problems. Although many fuzzy graph problems are NP-complete, computationally effective algorithms do exist—we have presented several here. We plan to address further computational questions in the future and hope to develop software suited to the analysis of fuzzy graphs.

VIII. Acknowledgments

This work was supported by the U.S. Department of Energy.

IX. Appendix: Fuzzy Arithmetic

A. Fuzzy Numbers and Arithmetic

1. Fuzzy Numbers

A fuzzy subset A built on $\Re$, is a *fuzzy number* if A is normalized and for each $\alpha \in (0,1]$, the α -cut S_α is convex. Since each S_α is a subset of the real line, S_α is convex if and only if S_α is an interval. This convexity requirement reduces to the area between the *x*-axis and the graph of μ_A being convex.

2. Arithmetic

Arithmetic operations are defined by Equation (12). However, another way to formulate the arithmetic is to use α -cuts and then perform the operation with the resulting intervals. In this manner we define:

(93) $$(A \otimes B)_\alpha = A_\alpha \otimes B_\alpha = \left\{ z \in \Re \,\middle|\, \exists a \in A_\alpha \text{ and } \exists b \in B_\alpha \text{ with } a \otimes b = z \right\},$$

then the resulting fuzzy set $A \otimes B$ has membership equation:

(94) $$\tilde{\mu}_{A \otimes B}(z) = \max\left\{ \alpha \,\middle|\, z \in (A \otimes B)_\alpha \right\}.$$

Claim: The max/min definition of Equation (12) and the α -cut definition of arithmetic operations are equivalent. Formally :

(95) $$\tilde{\mu}_{A \otimes B}(z) = \max\left\{ \alpha \,\middle|\, z \in (A \otimes B)_\alpha \right\} = \max_{x \otimes y = z}\left\{ \min\left\{ \mu_A(x), \mu_B(y) \right\} \right\} = \mu_{A \otimes B}(z) .$$

> **Proof:** Suppose $z \in (A \otimes B)_\alpha$. Then $\exists a \in A_\alpha$ and $\exists b \in B_\alpha$ such that $a \otimes b = z$. Thus, we have $\mu_A(a) \geq \alpha$ and $\mu_B(b) \geq \alpha$, so that $\min\{\mu_A(a), \mu_B(b)\} \geq \alpha$, which implies $\max_{x \otimes y = z}\left\{ \min\left\{ \mu_A(x), \mu_B(y) \right\} \right\} \geq \alpha$. Since α was arbitrary, we can now conclude $\max_{x \otimes y = z}\left\{ \min\left\{ \mu_A(x), \mu_B(y) \right\} \right\} \geq \max\left\{ \alpha \,\middle|\, z \in (A \otimes B)_\alpha \right\}$. Now let $\tilde{\alpha} = \max\left\{ \alpha \,\middle|\, z \in (A \otimes B)_\alpha \right\}$, so again $\exists a \in A_{\tilde{\alpha}}$ and $\exists b \in B_{\tilde{\alpha}}$ such that $a \otimes b = z$. However, since $\tilde{\alpha}$ is maximal there do not

exist a' and b' such that $a' \otimes b' = z$ with $\mu_A(a') > \tilde{\alpha}$ and $\mu_B(b') > \tilde{\alpha}$. Thus, for all x and y such that $x \otimes y = z$, $\min\{\mu_A(x), \mu_B(y)\} \le \tilde{\alpha}$. Hence,

$$\max_{x \otimes y = z} \{\min\{\mu_A(x), \mu_B(y)\}\} \le \tilde{\alpha} = \max\{\alpha \mid z \in (A \otimes B)_\alpha\}.$$

We can now conclude:

$$\tilde{\mu}_{A\otimes B}(z) = \max\{\alpha \mid z \in (A \otimes B)_\alpha\} = \max_{x \otimes y = z} \{\min\{\mu_A(x), \mu_B(y)\}\} = \mu_{A\otimes B}(z).$$

Thus, the two definitions of arithmetic operations are equivalent.

3. Closure

Fuzzy numbers are closed under the arithmetic operations. If all α -cuts of A and B are convex, then all α -cuts are intervals: $A_\alpha = [a_1^\alpha, a_2^\alpha]$ and $B_\alpha = [b_1^\alpha, b_2^\alpha]$. From the equivalence of the two definitions of arithmetic operations the following hold:

(96a) $$(A+B)_\alpha = A_\alpha + B_\alpha = [a_1^\alpha + b_1^\alpha, a_2^\alpha + b_2^\alpha],$$

(96b) $$(A-B)_\alpha = A_\alpha - B_\alpha = [a_1^\alpha - b_2^\alpha, a_2^\alpha - b_1^\alpha],$$

(96c) $$(A \times B)_\alpha = A_\alpha \times B_\alpha = \begin{cases} [a_1^\alpha \times b_1^\alpha, a_2^\alpha \times b_2^\alpha] & \text{if } A_\alpha \ge 0 \text{ and } B_\alpha \ge 0 \\ [a_2^\alpha \times b_2^\alpha, a_1^\alpha \times b_1^\alpha] & \text{if } A_\alpha \le 0 \text{ and } B_\alpha \le 0 \\ [a_1^\alpha \times b_2^\alpha, a_2^\alpha \times b_1^\alpha] & \text{if } A_\alpha \le 0 \text{ and } B_\alpha \ge 0 \\ [a_1^\alpha \times b_2^\alpha, a_2^\alpha \times b_2^\alpha] & \text{if } a_1^\alpha < 0 < a_2^\alpha \text{ and } B_\alpha \ge 0 \\ [a_2^\alpha \times b_1^\alpha, a_1^\alpha \times b_1^\alpha] & \text{if } a_1^\alpha < 0 < a_2^\alpha \text{ and } B_\alpha \le 0 \\ [\max(a_1^\alpha \times b_2^\alpha, a_2^\alpha \times b_1^\alpha), \max(a_1^\alpha \times b_1^\alpha, a_2^\alpha \times b_2^\alpha)] & \text{if } a_1^\alpha < 0 < a_2^\alpha \text{ and } b_1^\alpha < 0 < b_2^\alpha \end{cases},$$

(96d) $$(A \div B)_\alpha = A_\alpha \div B_\alpha = \begin{cases} [a_1^\alpha \div b_2^\alpha, a_2^\alpha \div b_1^\alpha] & \text{if } A_\alpha > 0 \text{ and } B_\alpha > 0 \\ [a_2^\alpha \div b_1^\alpha, a_1^\alpha \div b_2^\alpha] & \text{if } A_\alpha < 0 \text{ and } B_\alpha < 0 \\ [a_1^\alpha \div b_1^\alpha, a_2^\alpha \div b_2^\alpha] & \text{if } A_\alpha < 0 \text{ and } B_\alpha > 0 \\ [a_2^\alpha \div b_2^\alpha, a_1^\alpha \div b_1^\alpha] & \text{if } A_\alpha > 0 \text{ and } B_\alpha < 0 \\ \text{undefined} & \text{otherwise} \end{cases},$$

(96e) $$(\min(A,B))_\alpha = \min(A_\alpha, B_\alpha) = [\min(a_1^\alpha, b_1^\alpha), \min(a_2^\alpha, b_2^\alpha)],$$

(96f) $$(\max(A,B))_\alpha = \max(A_\alpha, B_\alpha) = [\max(a_1^\alpha, b_1^\alpha), \max(a_2^\alpha, b_2^\alpha)].$$

Thus, if each α -cut of A and B is convex then all α-cuts of $A \otimes B$ are convex for $\otimes \in \{+,-,\times,\div,\min,\max\}$. Also if A and B are normalized, then $\exists a \in \mathrm{supp}(A)$ and $\exists b \in \mathrm{supp}(B)$ such that $\mu_A(a) = 1$ and $\mu_B(b) = 1$. Thus, by definition $\mu_{A\otimes B}(a \otimes b) = 1$. We can now conclude that if A and B are fuzzy numbers, then so is $A \otimes B$. Thus, the set of fuzzy numbers is closed under all arithmetic operations.

4. Elementary Properties

It is easy to see that addition, multiplication, minimum, and maximum are commutative and associative. The distributive property also holds.

Proof:

$$\mu_{A\times(B+C)}(z) = \max_{x\times y=z}\left\{\min\{\mu_A(x),\mu_{B+C}(y)\}\right\} = \max_{x\times y=z}\left\{\min\left\{\mu_A(x),\max_{u+v=y}\{\min\{\mu_B(u),\mu_C(v)\}\}\right\}\right\}$$
$$= \max_{x\times y=z}\left\{\max_{u+v=y}\left\{\min\{\mu_A(x),\mu_B(u),\mu_C(v)\}\right\}\right\} = \max_{x\times(u+v)=z}\left\{\min\{\mu_A(x),\mu_B(u),\mu_C(v)\}\right\} =$$
$$\max_{(x\times u)+(x\times v)=z}\left\{\min\{\mu_A(x),\mu_B(u),\mu_C(v)\}\right\} = \max_{(x\times u)+(x\times v)=z}\left\{\min\{\min\{\mu_A(x),\mu_B(u)\},\min\{\mu_A(x),\mu_C(v)\}\}\right\}$$
$$= \max_{a+b=z}\max_{x\times u=a}\max_{x\times v=b}\left\{\min\{\min\{\mu_A(x),\mu_B(u)\},\min\{\mu_A(x),\mu_C(v)\}\}\right\} =$$
$$\max_{a+b=z}\left\{\min\left\{\max_{x\times u=a}\{\min\{\mu_A(x),\mu_B(u)\}\},\max_{x\times v=b}\{\min\{\mu_A(x),\mu_C(v)\}\}\right\}\right\}$$
$$= \max\left\{\min\left(\mu_{A\times B}(a),\mu_{A\times C}(b)\right)\right\} = \mu_{(A\times B)+(A\times C)=z}(z).$$

5. Inverses and Deconvolution

The field structure of fuzzy numbers fails when addressing the existence of additive and multiplicative inverses. It is clear that the additive and multiplicative identities are the crisp numbers 0 and 1 respectively. Thus, ideally we desire $A+(-A)=0$, for all fuzzy numbers A. Unfortunately, this is not the case.

Proof:

Denote the length of the support interval of a fuzzy number B by $\left|\text{supp}(B)\right|$. Using the α -cut definition of addition we obtain

$$\left(A+(-A)\right)_\alpha = A_\alpha + (-A)_\alpha = \left[a_\alpha^-, a_\alpha^+\right] + \left[-a_\alpha^+, -a_\alpha^-\right] = \left[a_\alpha^- - a_\alpha^+, a_\alpha^+ - a_\alpha^-\right].$$

Thus, the length of each α -cut is $\left|\left(A+(-A)\right)_\alpha\right| = \left(a_\alpha^+ - a_\alpha^-\right) - \left(a_\alpha^- - a_\alpha^+\right) = 2a_\alpha^+ - 2a_\alpha^-$. Taking the limit as $\alpha \to 0$, one finds that

$$\left|\text{supp}\left(A+(-A)\right)\right| = 2(\sup\left\{\text{supp}(A)\right\} - \inf\left\{\text{supp}(A)\right\}) = 2\left|\text{supp}(A)\right|.$$

Hence, $\left(A+(-A)\right)=0$ if and only if A is crisp. Thus $-A$ is not an additive inverse for A. In fact, for all fuzzy numbers A and B, $\left|\text{supp}(A+B)\right| \geq \left|\text{supp}(A)\right|$ and $\left|\text{supp}(A+B)\right| \geq \left|\text{supp}(B)\right|$ allowing only crisp numbers to have inverses.

A similar problem occurs when searching for multiplicative inverses. The main problem is that performing arithmetic operations on fuzzy numbers increases the fuzziness. That is, the support of $A\otimes B$ is larger than the support of A or B, if both A and B are fuzzy. This makes it impossible to solve equations of the type:

(97) $$A\otimes X = C,$$

provided

(98) $$\left|\text{supp}(C)\right| < \left|\text{supp}(A)\right|.$$

On the other hand, if $\left|\text{supp}(C)\right| > \left|\text{supp}(A)\right|$, the equations $A+X=C$, $A-X=C$, or $A\times X=C$ are solvable by a process called *deconvolution.* This process is accomplished using α -cuts. Let us consider the equation: $A + X = C$. Let $A_\alpha = \left[a_1^\alpha, a_2^\alpha\right]$ and $C_\alpha = \left[c_1^\alpha, c_2^\alpha\right]$. The goal is now to find an interval $X_\alpha = \left[x_1^\alpha, x_2^\alpha\right]$ such that $A_\alpha + X_\alpha = C_\alpha$. Now that the problem has been reduced to intervals, the solution is clear:

(99) $$x_1^\alpha = c_1^\alpha - a_1^\alpha \text{ and } x_2^\alpha = c_2^\alpha - a_2^\alpha$$

Now define $\mu_X(y) = \max\{\alpha | y \in X_\alpha\}$. From the equivalence of arithmetic by α -cuts and arithmetic using Equation (12), we see that in fact $A + X = C$ has solution X, where X is fuzzy with membership equation given by:

(100) $$\mu_X(y) = \max\{\alpha | y \in X_\alpha\}.$$

The equation $A \otimes X = C$ with other operations follows similarly provided the supports satisfy the above Equation (98) and in the case of division the following also holds: if $0 \in \operatorname{supp}(C)$, then $0 \in \operatorname{supp}(A)$.

B. Specific representations

Using the general fuzzy number representation for calculations can be time-consuming and tedious in many cases. Below we review several simple forms of the membership function for a fuzzy number.

1. Interval Representation

In the *interval representation* a fuzzy number $A = [A_0, A_1]$ has a membership function [KG 85]:

(101) $$\mu_A(x) = \begin{cases} 1, & x \in [A_0, A_1] \\ 0, & x \notin [A_0, A_1] \end{cases}$$

There are simple formulas for the basic arithmetic operations:

(102a) $$[A_0, A_1] + [B_0, B_1] = [A_0 + B_0, A_1 + B_1],$$

(102b) $$[A_0, A_1] - [B_0, B_1] = [A_0 - B_1, A_1 - B_0],$$

(102c) $$[A_0, A_1] \times [B_0, B_1] = [A_0 \times B_0, A_1 \times B_1],$$

(102d) $$[A_0, A_1] \div [B_0, B_1] = [A_0 \div B_1, A_1 \div B_0],$$

(102e) $$\min\{[A_0, A_1], [B_0, B_1]\} = [\min\{A_0, B_0\}, \min\{A_1, B_1\}],$$

(102f) $$\max\{[A_0, A_1], [B_0, B_1]\} = [\max\{A_0, B_0\}, \max\{A_1, B_1\}].$$

2. Triangular Representation

In the *triangular representation* a fuzzy number A = [A_0,A_1,A_2] has a membership function [KG 85]:

(103)
$$\mu_A(x) = \begin{cases} \dfrac{x - A_0}{A_1 - A_0}, & x \in [A_0, A_1] \\ \dfrac{A_2 - x}{A_2 - A_1}, & x \in [A_1, A_2]. \\ 0, & x \notin [A_0, A_2] \end{cases}$$

Here also there are simple formulas for the basic arithmetic operations:

(104a) $$[A_0, A_1, A_2] + [B_0, B_1, B_2] = [A_0 + B_0, A_1 + B_1, A_2 + B_2],$$

(104b) $$[A_0, A_1, A_2] - [B_0, B_1, B_2] = [A_0 - B_2, A_1 - B_1, A_2 - B_0],$$

(104c) $$[A_0, A_1, A_2] \times [B_0, B_1, B_2] = [A_0 \times B_0, A_1 \times B_1, A_2 \times B_2],$$

(104d) $$[A_0, A_1, A_2] \div [B_0, B_1, B_2] = [A_0 \div B_2, A_1 \div B_1, A_2 \div B_0],$$

(104e) $$\min\{[A_0, A_1, A_2],[B_0, B_1, B_2]\} = [\min\{A_0, B_0\}, \min\{A_1, B_1\}, \min\{A_2, B_2\}],$$

(104f) $$\max\{[A_0, A_1, A_2],[B_0, B_1, B_2]\} = [\max\{A_0, B_0\}, \max\{A_1, B_1\}, \max\{A_2, B_2\}].$$

X. References

[Ad 80] J. M. Adamo, "Fuzzy Decision Trees," *Fuzzy Sets and Systems*, **4** (1980) 207-219.

[BD 85] G. Bortolan and R. Degani, "A Review of Some Methods for Ranking Fuzzy Subsets," *Fuzzy Sets and Systems*, **15** (1985) 1-19.

[Ch 85] S.-H. Chen, "Ranking Fuzzy Numbers with Maximizing Set and Minimizing Set," *Fuzzy Sets and Systems*, **17** (1985) 113-129.

[CK 81] S. Chanas and J. Kamburowski, "The Use of Fuzzy Variables in PERT," *Fuzzy Sets and Systems*, **5** (1981) 11-19.

[CK 82] S. Chanas and W. Kolodziejczyk, "Maximum Flow in a Network with Fuzzy Arc Capacities," *Fuzzy Sets and Systems*, **8** (1982) 165-173.

[CK 84] S. Chanas and W. Kolodziejczyk, "Real-Valued Flows in a Network with Fuzzy Arc Capacities," *Fuzzy Sets and Systems*, **13** (1984) 139-151.

[CK 86] S. Chanas and W. Kolodziejczyk, "Integer Flows in Network with Fuzzy Capacity Constraints," *Networks*, **16** (1986) 17-31.

[DP 83] D. Dubois, and H. Prade, "Ranking Fuzzy Numbers in the Setting of Possibility Theory," *Information Sciences*, **30** (1983) 183-224.

[DVV 88] M. Delgado, J. L. Verdegay, and M. A. Vila, "A Procedure for Ranking Fuzzy Numbers using Fuzzy Relations," *Fuzzy Sets and Systems*, **26** (1988) 49-62.

[Gi 85] A. Gibbons, *Algorithmic Graph Theory*, (Cambridge: Cambridge University Press, 1985).

[GJ 79] M. Garey and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, (New York: W. H. Freeman and Company, 1979).

[IN 84] H. Itakura and Y. Nishikawa, “Fuzzy Network Technique for Technological Forecasting,” *Fuzzy Sets and Systems*, **14** (1984) 99-113.

[Ka 86] A. Kandel, *Fuzzy Mathematical Techniques with Applications*, (Reading, Massachusetts: Addison-Wesley Publishing Company, 1986).

[KF 88] G. Klir and T. Folger, *Fuzzy Sets, Uncertainty, and Information*, (Edgewood Cliffs, New Jersey: Prentice Hall, 1988).

[KG 85] A. Kaufmann and M. Gupta, *Introduction to Fuzzy Arithmetic*, (New York: Van Nostrand Reinhold Company, 1985).

[KR 82] K. H. Kim and F. W. Roush, “Fuzzy Flows on Networks,” *Fuzzy Sets and Systems*, **8** (1982) 35-38.

[Kl 91] C. M. Klein, “Fuzzy Shortest Paths,” *Fuzzy Sets and Systems*, **39** (1991) 27-41.

[La 76] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids*, (New York: Holt, Rhinehart and Winston, 1976).

[LC 93] K-C. Lin and M-S. Chern, “The Fuzzy Shortest Path Problem and Its Most Vital Arcs,” *Fuzzy Sets and Systems*, **58** (1993) 343-353.

[MR 96] R. Motwani and P. Raghavan, *Randomized Algorithms*, (New York: Cambridge University Press, 1996).

[OM 87] S. Ovchinnikov and M. Migdal, “On Ranking Fuzzy Sets,” *Fuzzy Sets and Systems*, **24** (1987) 113-116.

[Ov 89] S. V. Ovchinnikov, “Transitive Fuzzy Orderings of Fuzzy Numbers,” *Fuzzy Sets and Systems*, **30** (1989) 283-295.

[PJ 93] C-S. Peng and T. T-Y. Juang, “Fuzzy Network Flow,” *Congressus Numerantium*, **98** (1993) 156-162.

[Se 92] R. Sedgewick, *Algorithms in C++*, (Reading, Massachussetts: Addison-Wesley Publishing Company, 1992).

[Sh 79] D. R. Shier, “On Algorithms for Finding the *k* Shortest Paths in a Network,” *Networks*, **9** (1979) 195-214.

[To 88] D. M. Topkis, “A *k* Shortest Path Algorithm for Adaptive Routing in Communications Networks,” *IEEE Transactions on Communications*, **36** (1988) 855-859.

[Ye 71] J. Y. Yen, “Finding the *K* Shortest Loopless Paths in a Network,” *Management Science*, **17** (1971) 712-716.